AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Currently Amended) A method of distributing software from a server to a ~~client~~ computing device over a ~~computer~~ data network, said computing device having a web browser providing a cache, said method comprising:

deploying by ~~downloading~~sending, from said server to said ~~client~~ computing device via said computing device web browser, ~~an~~ at least one applet ~~including~~ providing instructions; and

executing said applet instructions with ~~that~~ the computing device to (i) create a cache ~~independent of said browser cache~~provide, on the ~~client~~ computing device that is independent of said browser cache, ~~a caching and class loading mechanism; and~~

~~using said caching and class loading mechanism to~~ (ii) retrieve additional resources over the data network from at least one remote source, (iii) store said retrieved resources into said applet-created cache, and ~~said applet instructions provided on the client and~~ (iv) load additional ~~applet~~ ~~modules~~ resources from said applet-created cache for use by said computing device~~on demand~~,

wherein said applet manages the contents of said applet-created cache and controls the persistence of said resources stored within said applet-created cache independently of said browser cache functionality.

2. (currently amended) A method as in claim 1 wherein said ~~client~~ computing device provides a particular virtual machine applet execution environment, said ~~downloading~~ deploying step comprises downloading an applet package that has been customized for said particular virtual machine applet execution environment, and said ~~using~~ executing step includes retrieving and loading additional applet modules not specific to said particular virtual machine applet execution environment.

3. (currently amended) A method as in claim 1 wherein said ~~client~~ computing device web browser comprises ~~provides~~ a particular web browser, said ~~downloading~~ deploying step comprises downloading an applet package that has been customized for said particular web browser, and said ~~using~~ executing step includes retrieving and loading additional applet modules not specific to said particular web browser.

4. (currently amended) A method as in claim 1 wherein said ~~downloading~~ deploying step includes packaging ~~said caching and class~~a resource loading mechanism within a stream and downloading said stream in a single http transaction.

5. (Previously presented) A method of distributing software from a server to a client over a computer network, said method comprising:

downloading, from said server to said client, an applet including a caching and class loading mechanism; and

using said caching and class loading mechanism to retrieve and load additional applet modules on demand,

wherein said downloading step includes packaging said caching and class loading mechanism within a stream and downloading said stream in a single http transaction, and

wherein said stream includes an uncompressed archive file containing said caching and class loading mechanism.

6. (Original) A method as in claim 5 wherein said stream includes at least one digital signature.

7. (currently amended) A method as in claim 1 wherein said ~~using~~ executing step includes verifying, under control of code within said applet, at least one further applet module downloaded on demand.

8. (currently amended) A method as in claim 1 further including constructing, on the ~~client~~computing device, a cache map ~~with said caching and class loading mechanism~~.

9. (currently amended) A method as in claim 1 wherein said ~~using~~ executing step includes loading at least further applet module to ~~a~~ be persistent in said applet-created cache on the ~~client~~ computing device for subsequent use.

10. (currently amended) A method as in claim 1 wherein ~~said~~ additional applet modules are organized into plural functional modules, and said ~~using~~ executing step comprises downloading at least one of said functional modules on demand.

11. (currently amended) A method as in claim 1 wherein said ~~client~~ computing device includes a local non-volatile memory that persistently caches applet classes, and said ~~using~~ executing step includes requesting an applet class from said server conditioned on determining whether said applet class is already available in said non-volatile memory.

12. (currently amended) A method as in claim 1 wherein said computing device ~~client~~ includes a local non-volatile memory, and said ~~using~~ executing step includes determining whether a particular version of an applet class is stored in said non-volatile memory, and requesting said version from said server based on results of said determining step.

13. (currently amended) A method as in claim 1 wherein said ~~using~~ executing step includes persistently storing applet modules downloaded from said server to said computing device ~~client~~ in a local non-volatile memory associated with said computing device~~client~~.

14. (currently amended) A system for distributing software from a server to a ~~client~~ computing device over a ~~computer~~ data network, said computing device providing

a web browser having a cache, said system including a downloader that downloads, from said server ~~to said client~~ via the computing device web browser over said ~~computer~~ data network, ~~an~~ at least one applet ~~including~~ providing a caching and ~~class~~ resource loading mechanism, wherein said caching and ~~class~~ resource loading mechanism executes on said ~~client~~ computing device to create a cache distinct from said browser cache and to automatically retrieve and load additional ~~applet modules on demand~~ resources over said data network into said applet-created cache for use by said computing device,

wherein said applet manages the contents of said applet-created cache and controls the persistence of said downloaded resources within said applet-created cache, said persistence control being independent of browser cache persistence.

15. (currently amended) A system as in claim 14 wherein said computing device ~~client~~ provides a particular applet execution environment, said downloader downloads an applet package that has been customized for said particular applet execution environment, and said caching and loading mechanism retrieves and loads additional applet modules not specific to said particular applet execution environment.

16. (currently amended) A system as in claim 14 wherein said ~~client provides~~ web browser comprises a particular web browser, said downloader downloads an applet package that has been customized for said particular web browser, and said caching and loading mechanism retrieves and loads additional applet modules not specific to said particular web browser.

17. (Currently amended) A system as in claim 14 wherein said downloader packages said caching and ~~class~~ resource loading mechanism within a stream and downloads said stream within a single http transaction.

18. (Previously presented) A system for distributing software from a server to a client over a computer network, said system including a downloader that downloads, from said server to said client over said computer network, an applet including a caching and class loading mechanism, wherein said caching and class loading mechanism executes on said client to retrieve and load additional applet modules on demand,

wherein said downloader packages said caching and class loading mechanism within a stream and downloads said stream within a single http transaction, and

wherein said stream includes an uncompressed archive file containing said code representing caching and class loading mechanism.

19. (Original) A system as in claim 18 wherein said stream includes at least one digital signature.

20. (Original) A system as in claim 14 wherein said caching and loading mechanism verifies at least one further applet class downloaded on demand.

21. (Original) A system as in claim 14 wherein said caching and loading mechanism maintains a cache map.

22. (currently amended) A system as in claim 14 wherein said ~~client includes~~applet created cache comprises a persistent cache, and said caching and loading mechanism ~~loads~~ stores at least further applet class into said persistent cache for subsequent use.

23. (Original) A system as in claim 14 wherein said additional applet classes are organized into functional modules, and said using caching and loading mechanism downloads said functional modules on demand.

24. (currently amended) A system as in claim 14 wherein said computing device ~~client~~ includes a local non-volatile memory that persistently caches applet classes, and said caching and downloading mechanism requests an applet class from said server conditioned on determining whether said applet class is already available in said non-volatile memory.

25. (currently amended) A system as in claim 14 wherein said computing device ~~client~~ includes a local non-volatile memory, and said caching and loading mechanism determines whether a particular version of an applet class is stored in said non-volatile memory, and requests said version from said server based on results of said determining step.

26. (currently amended) A system as in claim 14 wherein said computing device ~~client~~ includes a local non-volatile memory associated therewith, and said caching and

loading mechanism persistently stores applet classes downloaded from said server to said computing device~~client~~.

27. (Currently amended) An applet comprising:

a first, platform-dependent stream including a ~~platform-dependent~~ caching mechanism; and

at least one further stream including at least one platform-independent functional ~~module~~resource,

wherein said platform-dependent stream ~~caching mechanism~~ is used to create a cache outside of a browser cache and to cache and load said platform-independent functional ~~module~~resources into said outside cache,

wherein said applet manages the contents of said outside cache and controls the persistence of said downloaded resources within said outside cache independently of browser cache persistence.

28. (Currently amended) A method of ~~downloading~~ deploying an ~~applet~~ application using a web browser having a cache comprising:

(a) downloading a first stream including a platform-dependent caching and loading mechanism and executing said first stream to create a cache different from said browser cache; and

(b) downloading and caching, into said stream-created cache using said platform-dependent caching and loading mechanism downloaded by step (b), at least one further stream including at least one platform-independent functional module,

wherein said first stream execution manages the contents of said stream-created cache and controls the persistence of said downloaded resources within said stream-created cache independently of web browser control of persistence of items stored in said browser cache.

29. (Currently amended) A system for downloading ~~an~~ at least one applet to an end user computing device having a web browser that maintains a web browser cache, said system comprising an applet virtual machine execution environment that receives, verifies and begins executing a first stream including a platform-dependent caching and loading mechanism, said caching and loading mechanism creating and maintaining a cache that is different from said browser cache, wherein said applet environment, under control of said platform-dependent caching and resource loading mechanism, requests, receives and persistently caches at least one further stream including at least one platform-independent functional module in said cache created by said received caching and resource loading mechanism,

wherein said caching and loading mechanism manages the contents of said different cache and controls the persistence of created resources within said different cache independently of persistence of items within said browser cache.

Cancel claims 30 and 31 without prejudice or disclaimer.